

INTERDISCIPLINARITATEA CURSURILOR INFORMATICE: PROGRAMAREA OPERAȚIILOR ARITMETICE ASUPRA NUMERELOR ÎN SISTEMUL BINAR

PAVEL Maria, PAVEL Dorin,

Universitatea de Sat din Tiraspol

Rezumat. *În articol se examinează posibilitatea implementării operațiilor aritmetice asupra numerelor în sistemul binar, care se studiază la cursul de Sisteme de Operare și Arhitectura Calculatoarelor, în programare, din punct de vedere metodologic.*

Cuvinte-cheie: *abordare interdisciplinară la informatică, metode didactice, operații aritmetice, sistem binar de numerație, programare.*

Abstract. *The article examines the possibility of implementing arithmetic operations on numbers in the binary system, which is studied in the course of Operating Systems and Computer Architecture, in programming, from a methodological point of view.*

Keywords: *interdisciplinary approach in computer science, teaching methods, arithmetic operations, binary numbering system, programming.*

Introducere

Programele universitare pentru specialitățile informatice includ un șir de cursuri, a căror conținut este foarte divers, însă cu tematici care oferă legături interdisciplinare fie pe orizontală (aceleași subiecte se studiază la diferite cursuri din diferite aspecte), fie pe verticală (noțiunile studiate la un curs sunt fundamentale pentru alte cursuri). Astfel, cursurile de studiere a unor limbaje de programare presupun interdisciplinaritate pe orizontală, deoarece structura unui limbaj de programare, tipurile de date, operatorii și instrucțiunile etc. diferă doar prin sintaxa și specificul abordării în limbajul respectiv. Interdisciplinaritatea pe verticală poate fi identificată între aceste cursuri și cursurile care presupun deja competențe de programare, dar care studiază algoritmi sub aspect tehnic și metodologic, așa cum ar fi: Metode Numerice, Tehnici de programare, Structuri de Date și Algoritmi etc.

Totuși, noțiunile fundamentale din cadrul cursului *Sisteme de Operare și Arhitectura Calculatoarelor* (SO și AC) sunt esențiale pentru toate celelalte discipline informatice studiate la specialitățile corespunzătoare, deoarece contribuie la conștientizarea proceselor interne ale mașinii, perceperea corectă a operațiilor logice ale procesorului în cazul apelării funcțiilor și executării instrucțiunilor din conținuturile curriculare ale acestora. Un subiect esențial din cadrul disciplinei SO și AC se referă la reprezentarea internă a diferitor tipuri de date, iar studierea superficială a acestuia, va contribui la formarea insuficientă a competențelor de programare, esențiale pentru orice specialist în domeniul informaticii. Orice program conține declarații de variabile de diferite tipuri de

date și operații asupra lor, iar conștientizarea proceselor interne de conversie a datelor din format extern în format intern și vice-versa, va contribui la evitarea erorilor logice de programare și implementarea celor mai eficiente tehnici din punct de vedere a costului de timp și memorie. Deoarece majoritatea cursurilor informatice recurg la implementarea algoritmilor sub formă de programe într-un limbaj sau altul, este evidentă importanța acestui subiect.

Datele de orice tip sunt reprezentate în format intern în sistemul binar de numerație, deoarece acesta este cel mai eficient mod de implementare din punct de vedere logic și tehnic, iar noțiuni introductive la acest subiect sunt incluse în curriculumul la informatică din clasa a șapte-a [1] chiar din capitolul 1 în care se studiază algoritmul de conversie a numerelor naturale din baza 10 în orice bază și invers. Se revine asupra sistemelor de numerație în clasa a 11-a [2], prin generalizarea algoritmului pentru numerele reale și studierea operațiilor asupra numerelor în binar: adunarea, scăderea, înmulțirea și împărțirea. Deci includerea subiectelor ce țin de sisteme de numerație și operațiile aritmetice asupra numerelor în baza 2 în curriculumul disciplinar la cursul SO și AC este indiscutabil necesară.

Unele aspecte ale reprezentării numerelor întregi în sistemul binar au fost analizate de autori în lucrarea [3], în care s-au abordat aspecte didactice ale implementării operatorilor logici pe biți. În lucrarea dată ne vom referi doar la operațiile aritmetice asupra numerelor binare.

Metode de cercetare

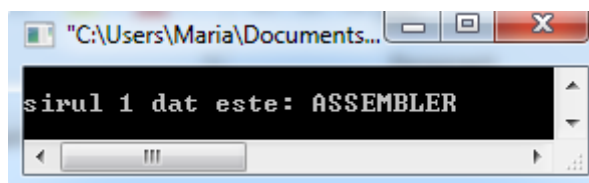
Mai sus s-a menționat că legăturile interdisciplinare între cursurile universitare informatice sunt indispensabile pregătirii viitorilor specialiști în domeniu, acestea identificându-se în mod evident între disciplinele de programare și cursul SO și AC. Astfel, conversia din sistemul binar în cel octal și hexazecimal și invers, sunt subiecte din acest curs, care contribuie la dezvoltarea competențelor cognitive formate în cadrul cursului de programare în limbajul C sau Assembler. Formatele de reprezentare a numerelor întregi: %d, %o și %x sau %X; consecutivitățile de conducere: \cifra_octala, \x,\X; utilizate în C sunt concepute de legătură cu SO și AC și Assembler, în care se lucrează cu adresele reprezentate în hexazecimal. De exemplu, șirul de caractere, ce reprezintă codurile hexazecimale ale unor simboluri ASCII:

```
char sir1[]="\x41\x53\x53\x45\x4D\x42\x4C\x45\x52";
```

poate fi afișat sub formă de text:

```
printf("sirul 1 dat este: %s\n",sir1);
```

obținând la ieșire:

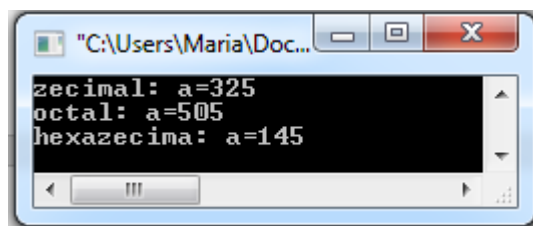


Iar afișarea unor date întregi în octal și hexazecimal se poate face astfel:

```
int a=325;
```

```
printf("\nzecimal: a=%d\noctal: a=%o\nhexazecima: a=%x\n",a,a,a);
```

obținând la ieșire:



```
zecimal: a=325
octal: a=505
hexazecima: a=145
```

Aceste fragmente de cod în C, pot fi utilizate la cursurile amintite și recomandate studenților pentru autoevaluarea corectitudinii efectuării unor conversii sau utilizării unor adrese.

Referitor la operațiile aritmetice asupra numerelor binare, pot fi aplicate diferite strategii didactice fie la disciplina SO și AC, fin la limbajul de programare C. Propunem următoarele două strategii:

1. La disciplina SO și AC se utilizează programe C, în care se dau numerele în baza 10, 8, sau 16, se convertesc în baza 2, apoi se realizează operația aritmetică dorită. În acest caz, programul oferit de profesor va servi ca un instrument adjuvant în verificarea corectitudinii conversiilor și operațiilor efectuate manual. Totodată, studenții vor avea un prim contact cu limbajul C, care se studiază într-un curs ulterior.

2. La cursul de programare (fie *Fundamentele Programării*, fie *Programare în Limbaje de Nivel Mediu*) se utilizează cunoștințele acumulate la SO și AC, se desfășoară un proiect de realizare a programului de conversie în baza 2 și de efectuare a operației aritmetice solicitate. Scenariul proiectului presupune împărțirea studenților în 4 grupe, care vor realiza separat fiecare operație aritmetică, conform tragerii la sorți. Condițiile minime asupra programului realizat de fiecare grup vor fi enunțate preventiv de către profesor: operația se va realiza prin intermediul unei funcții, limbajul de programare va fi la alegere sau în dependență de cursul la care se implementează proiectul, datele de intrare vor fi transmise funcției prin intermediul argumentelor sub formă de șir de caractere.

Desigur pot fi elaborate și implementate și alte metode care ar realiza interdisciplinaritatea cursurilor informatice, în dependență de creativitatea cadrului didactic.

Rezultate și discuții

Se cunoaște că fiecare programator își are stilul său, iar aceeași problemă poate fi soluționată prin diferite metode în dependență de restricțiile impuse. De exemplu, în cazul proiectului de mai sus studenții au realizat funcții de conversie din baza 10 în baza 2 cu ajutorul construirii șirului de

caractere rezultat simbol cu simbol ('1' sau '0'), utilizând restul împărțirii la 2 ($r=x\%2$) și partea întreagă ($x/=2$):

```
char* convert_bin(int x){
    int r,i,j;
    char s[255], rezultat[255];
    strcpy(rezultat,"");i=0;
    while(x){
        r=x%2;
        x/=2;
        if (r==0)s[i]='0'; else s[i]='1';
        i++;
    }
    while(i<16){
        s[i++]='0';
    }
    s[i]='\0';
    for(j=0,i=strlen(s)-1;i>=0;i--,j++)
        rezultat[j]=s[i];
    rezultat[j]='\0';
    return rezultat;
}
```

Acest cod a permis efectuarea următoarelor concluzii asupra cunoștințelor studenților:

- cunosc și pot implementa în program algoritmul de trecere din baza 10 în baza 2;
- cunosc structura unei funcții;
- pot realiza și implementa funcții ce întorc un șir de caractere;
- cunosc specificatorul de sfârșit de șir de caractere;
- implementează funcțiile de gestiune a șirurilor de caractere din fișierul header <string.h>.

Pentru adunarea a două numere în baza 2, studenții au elaborat următoarea funcție:

```
void summ_bin(char x[255], char y[255]){
    int r,i,n;
    char rezultat[255];
    r=0;
    i=strlen(x)-1;
    rezultat[i+1]='\0';
    while (i>=0){
```

```

        n=(int)x[i]+(int)y[i]-2*(int)'0'+r;
        rezultat[i--]='0'+n%2;
        r=n/2;
    }
    if(r==1)rezultat[0]='1';
    printf("suma\n%16s+\n%16s\n_____ \n%16s\n",x,y,rezultat);
}

```

Acest cod a permis efectuarea următoarelor concluzii asupra cunoștințelor studenților:

- cunosc și pot implementa în program algoritmul de adunare a două numere în baza 2;
- cunosc structura unei funcții;
- cunosc și pot utiliza operațiile aritmetice și cele de conversie de tip pentru a putea obține un număr pe baza caracterelor din șir;
- implementează operațiile aritmetice și cele de conversie de tip pentru a obține un caracter din șirul rezultat pe baza unui număr.

Dacă se vor apela cele două funcții în funcția principală, vom obține:

```

main()
{int m,n; char nx[255],mx[255];
    printf("m=");
    scanf("%d",&m);
    strcpy(mx,convert_bin(m));
    printf("%d in baza 2 este: %s\n",m,mx);
    printf("n=");
    scanf("%d",&n);
    strcpy(nx,convert_bin(n));
    printf("%d in baza 2 este: %s\n",n,nx);
    summ_bin(mx,nx);
}

```

Datele de ieșire vor fi:

```

"C:\Users\Maria\Documents\C-Free\Temp\zece_bin...
m=27      27 in baza 2 este: 00000000000011011
n=124     124 in baza 2 este: 00000000001111100
SUMA
00000000000011011+
00000000001111100
-----
00000000010010111
Press any key to continue . . .

```

Exemplele de mai sus pot fi realizate și cu ajutorul funcțiilor `strcpy()` și `strcat()` sau implementând alți algoritmi optimizați, în dependență de contingentul de studenți, pregătirea lor inițială, creativitatea și nivelul de dezvoltare a competențelor de programare.

Concluzii. Desfășurarea unor astfel de activități la disciplinele informatice vor contribui la o bună cunoaștere a unor procese interne ale mașinii, la însușirea unor algoritmi matematici și de programare referitor la conversii și operații în binar, la dezvoltarea competențelor de colaborare, lucru în echipă și de creativitate.

Articol realizat în cadrul proiectului de cercetări științifice „Metodologia implementării TIC în procesul de studiere a științelor reale în sistemul de educație din Republica Moldova din perspectiva inter/transdisciplinarității (concept STEAM)”, inclus în „Program de stat” (2020-2023), Prioritatea IV: Provocări societale, cifrul 20.80009.0807.20, cu suportul financiar oferit de Agenția Națională pentru Dezvoltare și Cercetare

Bibliografie

1. Gremalschi, A.; Vasilache, G.; Gremalschi, L. Informatica: Manual pentru clasa a 7-a. [Ed. a 3-a]. Ch.: I.E.P. Știința, 2020 (Tipogr. „Balacron”). 152 p. ISBN 978-9975-85-248-7.
2. Gremalschi, A.; Mocanu, Iu.; Gremalschi, L. Informatica: Manual pentru clasa a 11-a. Ch.: I.E.P. Știința, 2020 (Combinatul Poligrafic). 224 p. ISBN 978-9975-85-246-3.
3. Pavel, M.; Pavel, D. Abordări didactice ale aplicabilității operatorilor logici pe biți. In: *Materialele Conferinței Republicane a Cadrelor Didactice. Didactica științelor exacte*. Vol. 1, 27-28 februarie 2021. Chișinău, Republica Moldova: Tipografia Universității de Stat din Tiraspol, 2021, pp. 147-152. ISBN 978-9975-76-324-0.

CZU:37.853(076.5)

OPTICĂ ONDULATORIE – LUCRĂRI DE LABORATOR

PETCU Steliana, director, profesor

Școala Gimnazială Luceafărul, București, România

Rezumat. *Difracția este ansamblul fenomenelor produse de natura ondulatorie a luminii, care apar atunci când se propagă într-un mediu cu caracteristici eterogene foarte pronunțate. În articol se descriu trei lucrări de laborator la tema difracției, și anume:*

Laboratorul 1: Determinarea lungimii de undă a radiației folosite cu ajutorul rețelei de difracție

Laboratorul 2: Determinarea lărgimii fantei cu ajutorul difracției pe o fantă în lumina paralelă

Laboratorul 3: Determinarea distanței dintre fantele unui dispozitiv Young